**Ronaldlee Ejalu**

**CSC 555: Mining Big Data**

Project, Phase 2 (due Tuesday, November 23rd)

In this part of the project, you will execute queries using Hive, Pig and Hadoop streaming and develop a custom version of KMeans clustering. The schema and data is available at:

<http://cdmgcsarprd01.dpu.depaul.edu/CSC555/SSBM1/>

You should use your 3-node cluster for the final. Please be sure to submit all code. You should also submit the command lines you use and a screenshot of a completed run (just the last page, do not worry about capturing the entire output).

**I highly recommend creating a small sample input** (e.g., by running head -n 1000 lineorder.tbl > lineorder.tbl.sample, you can create a small version of lineorder with 1000 lines) and testing your code with a smaller file until you can verify that it works.

# Part 1: Pig

Implement the following query using Pig:

select c\_nation, AVG(lo\_extendedprice) as AVGL

from customer, lineorder

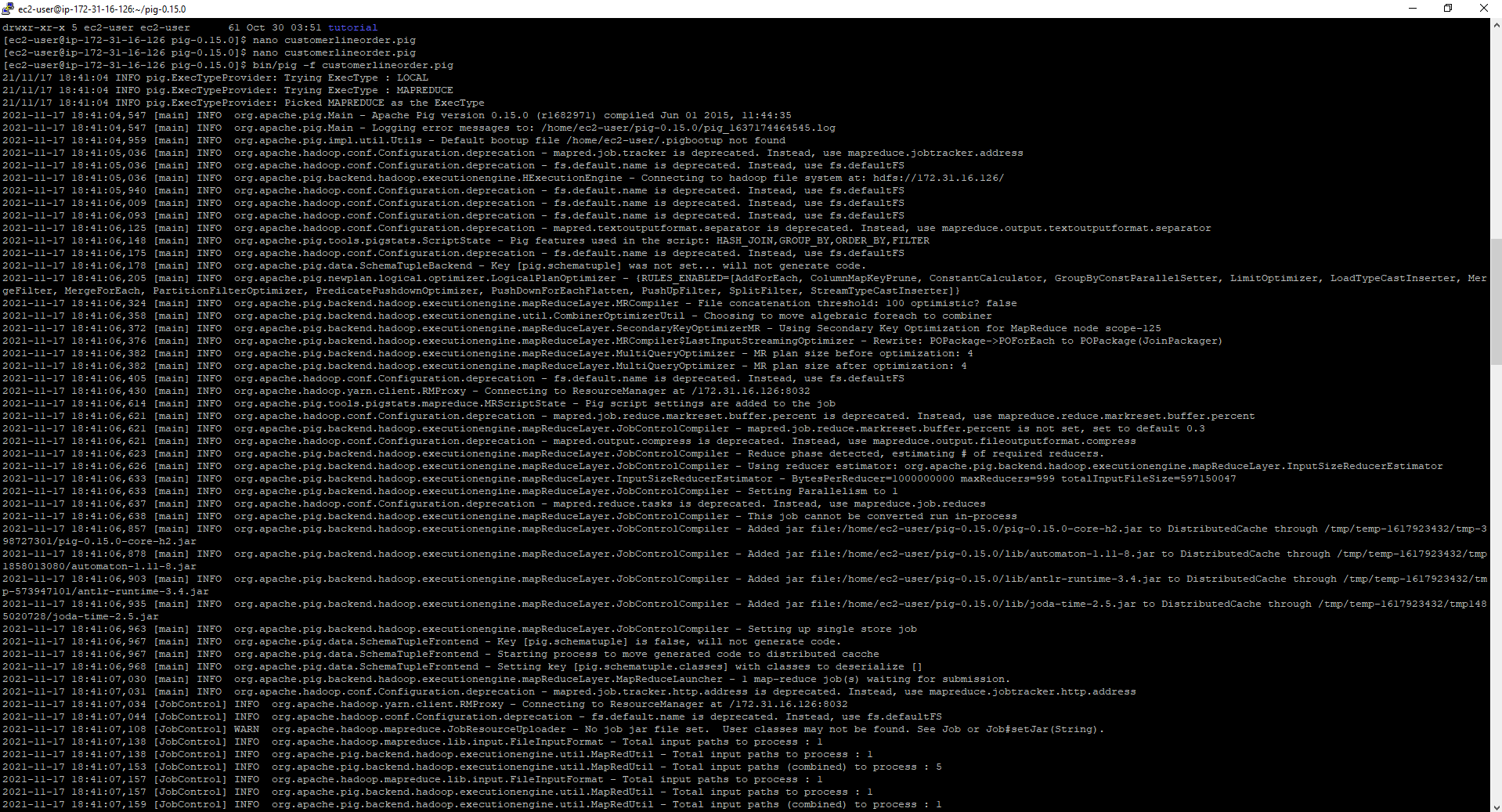
where lo\_custkey = c\_custkey

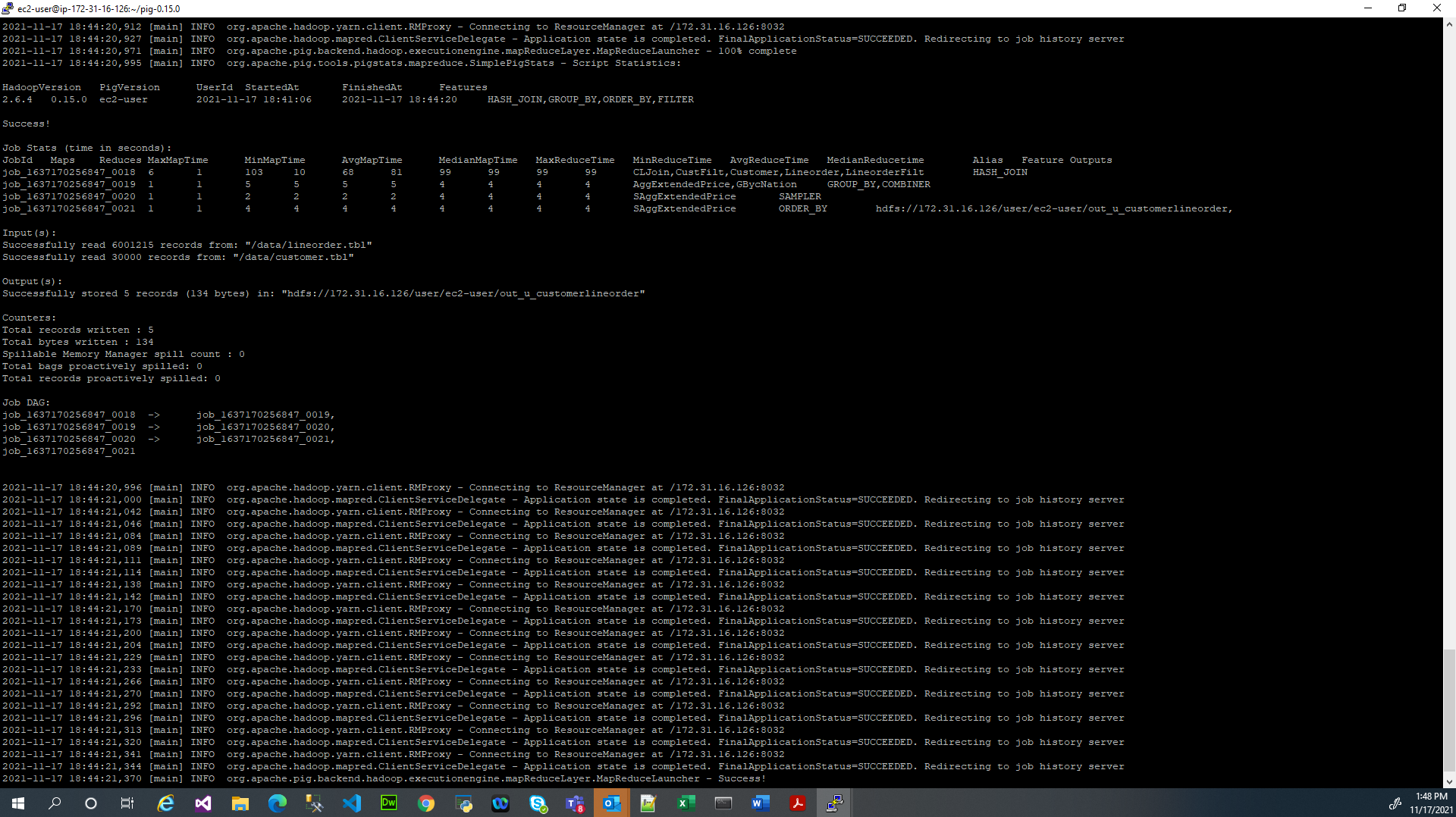
and c\_region = 'AFRICA'

and lo\_discount = 6 OR lo\_discount > 8

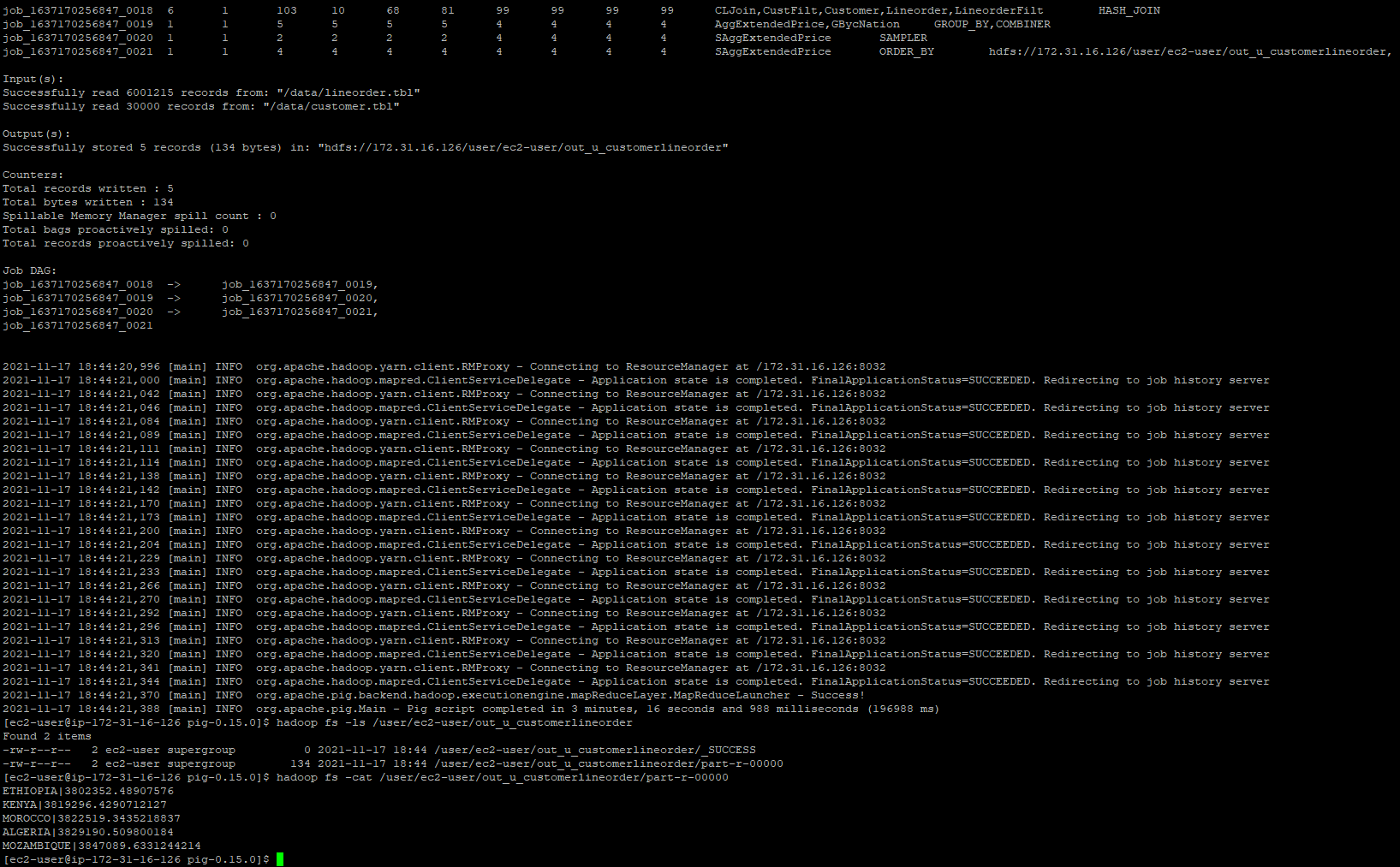
group by c\_nation

order by AVGL;





The output produced by the aggregation:



The pig script used:



# Part 2: Hadoop streaming

Implement the following query using Hadoop streaming:

select sum(lo\_revenue), d\_year, p\_brand1

from lineorder, dwdate, part

where lo\_orderdate = d\_datekey

and lo\_partkey = p\_partkey

and d\_sellingseason = 'Fall'

and p\_brand1 between 'MFGR#2121'

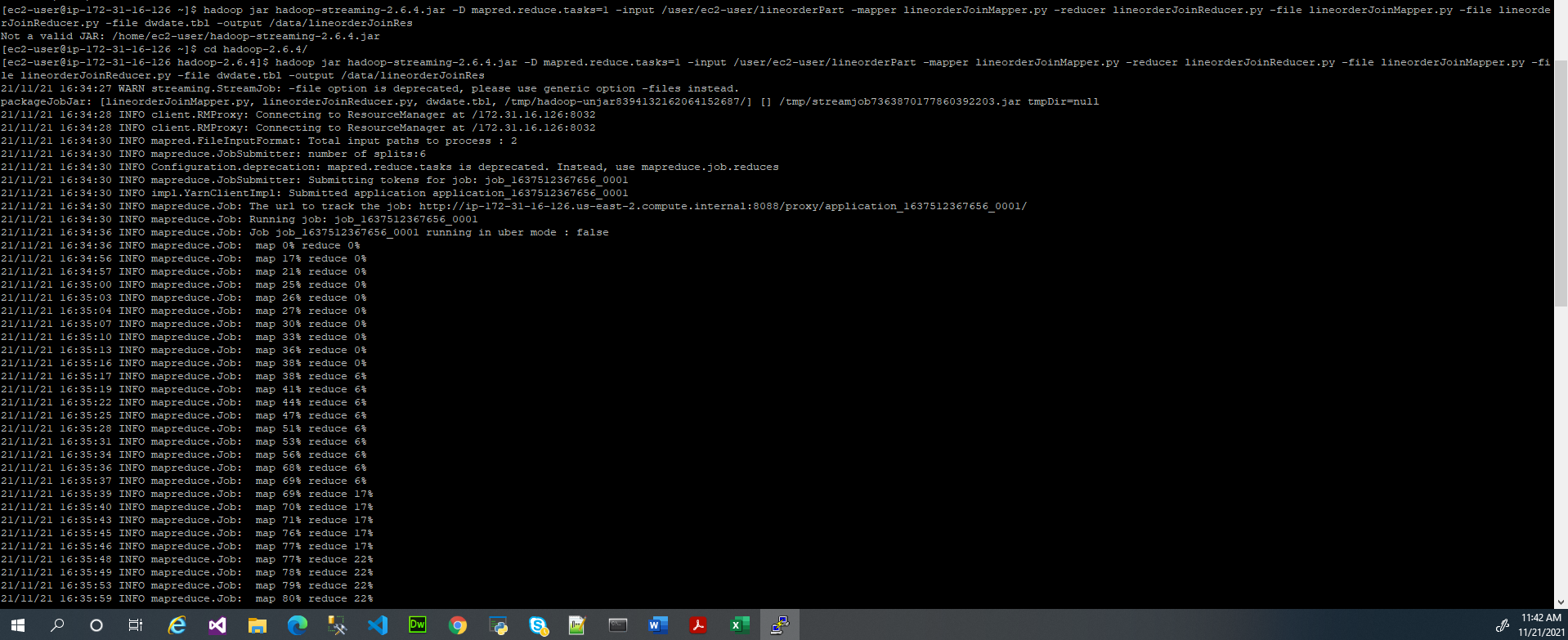
and 'MFGR#2138'

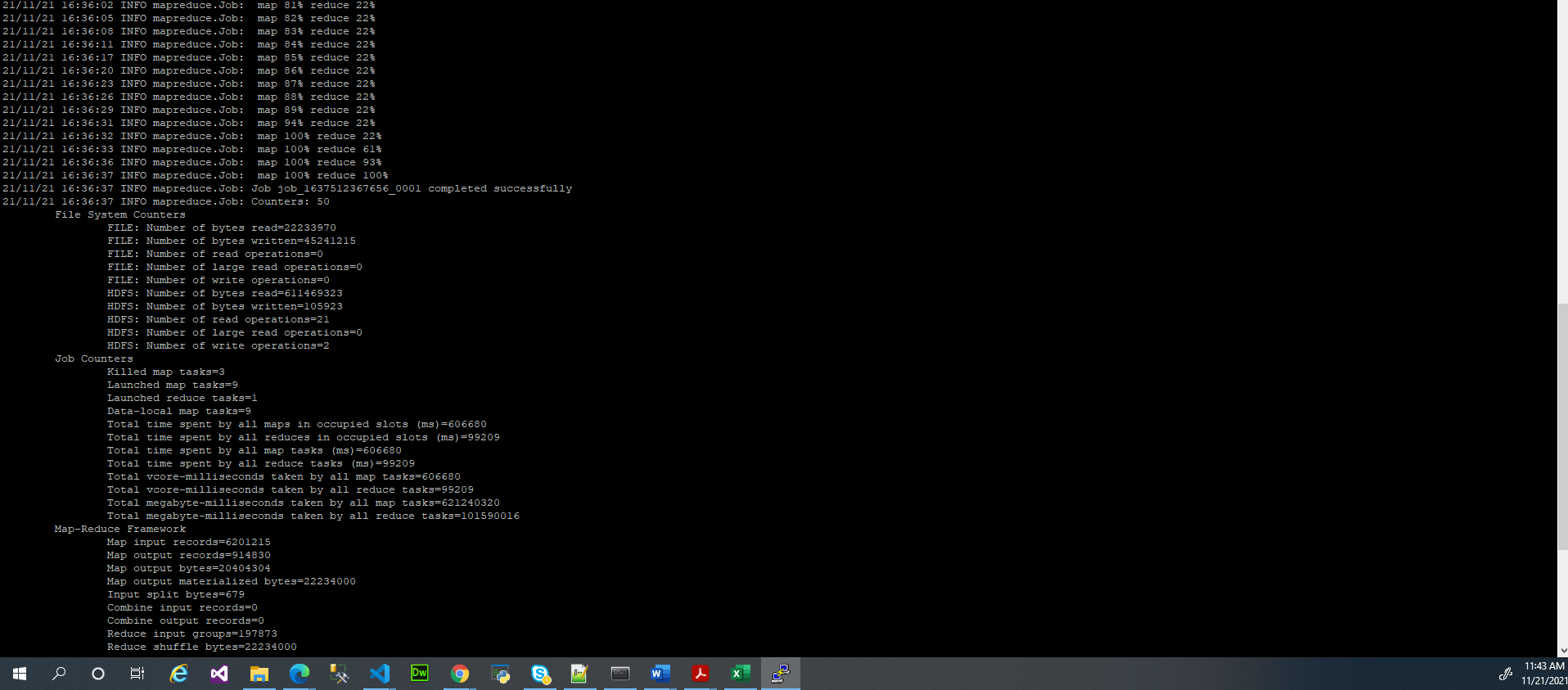
group by d\_year, p\_brand1

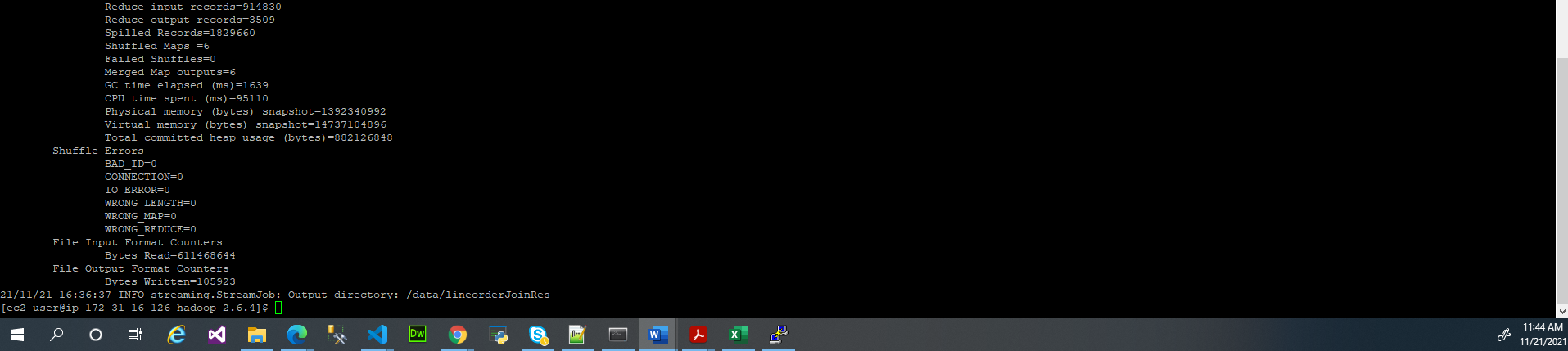
In Hadoop streaming, this will use a total of 3 passes (two joins and another one for GROUP BY). You can also choose to perform a map-side join with dwdate (only dwdate), which would result in a total of 2 passes (one join and one for GROUP BY).

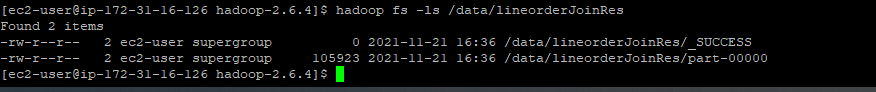
**Using map-side join with dwdate.**

**The first pass:**









**The code of the first pass:**

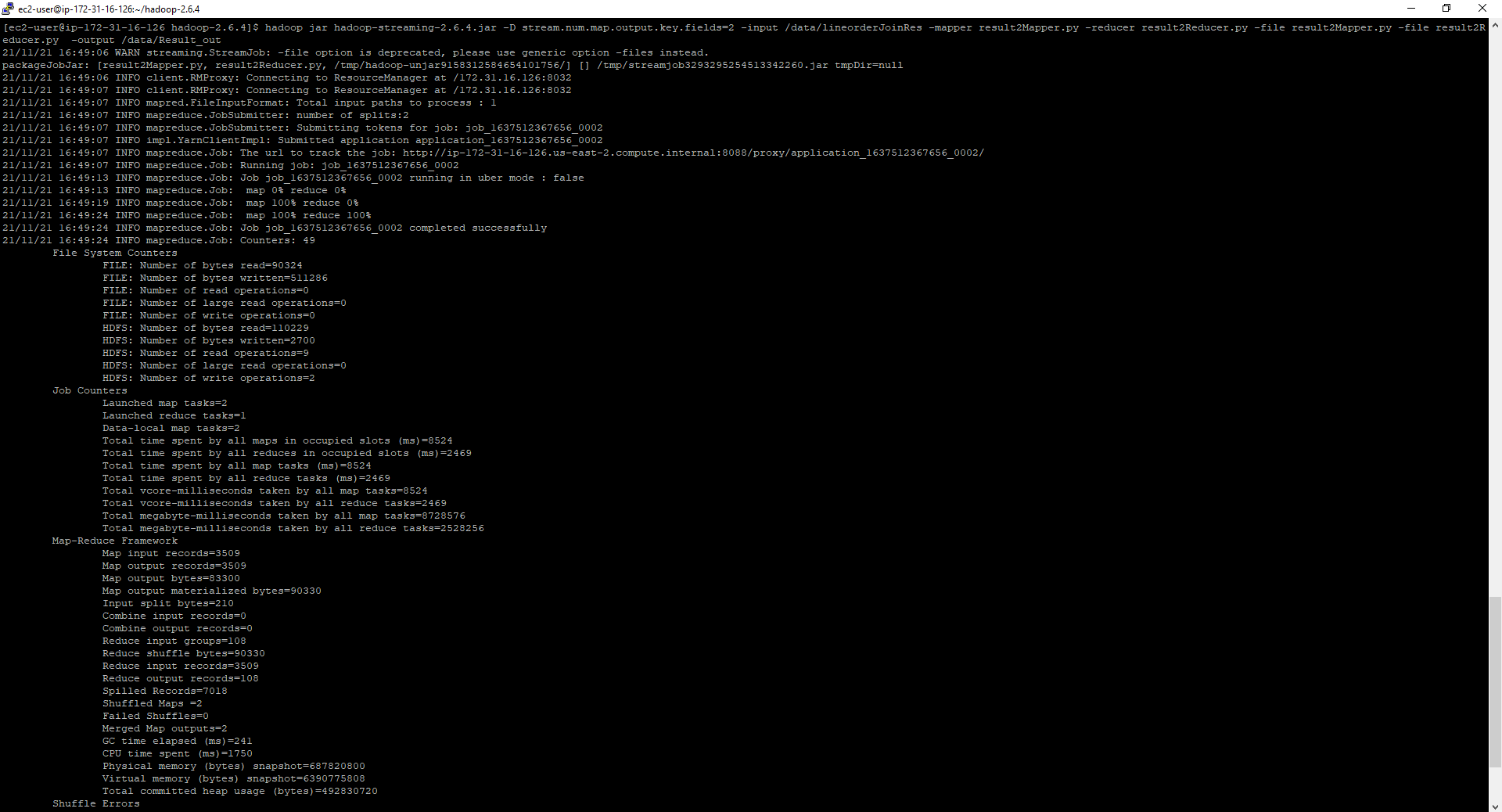
**lineorderJoinMapper.py**

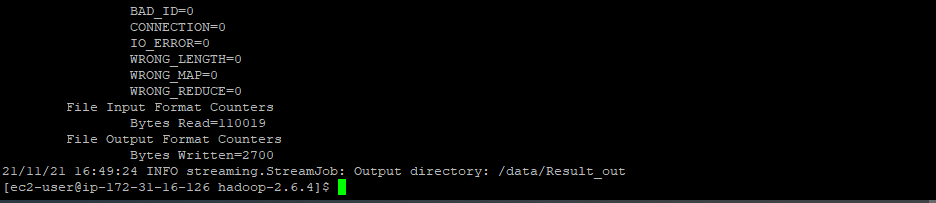


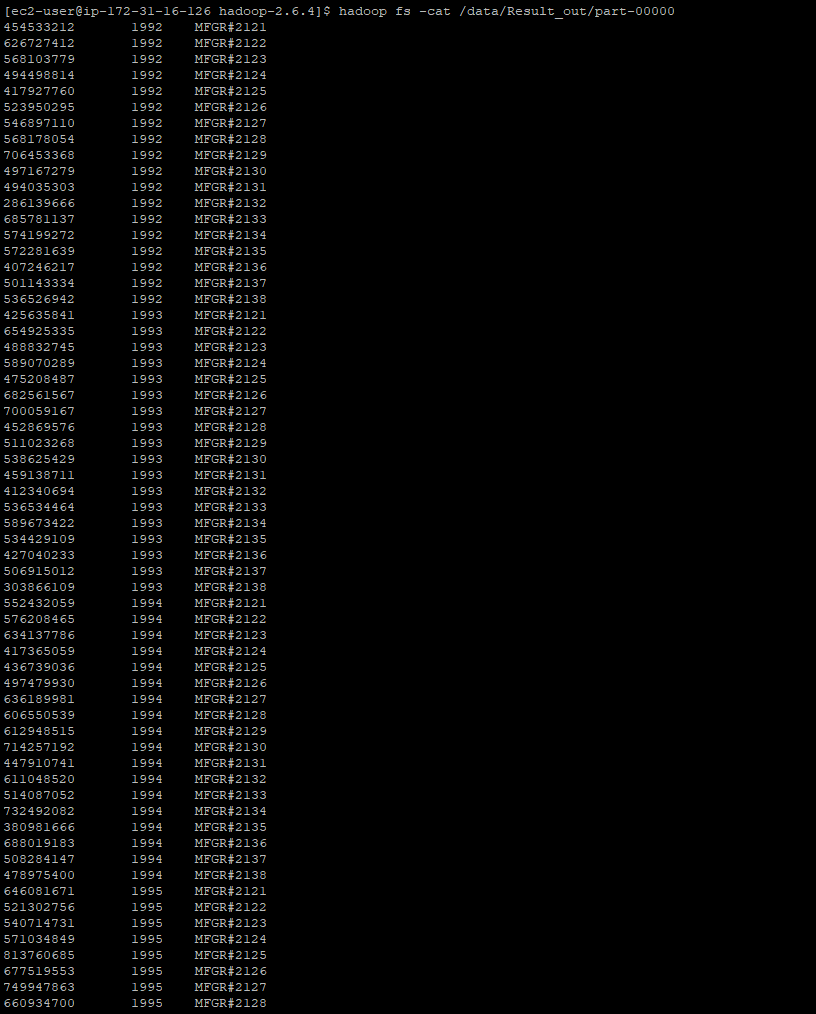
**lineorderJoinReducer.py**

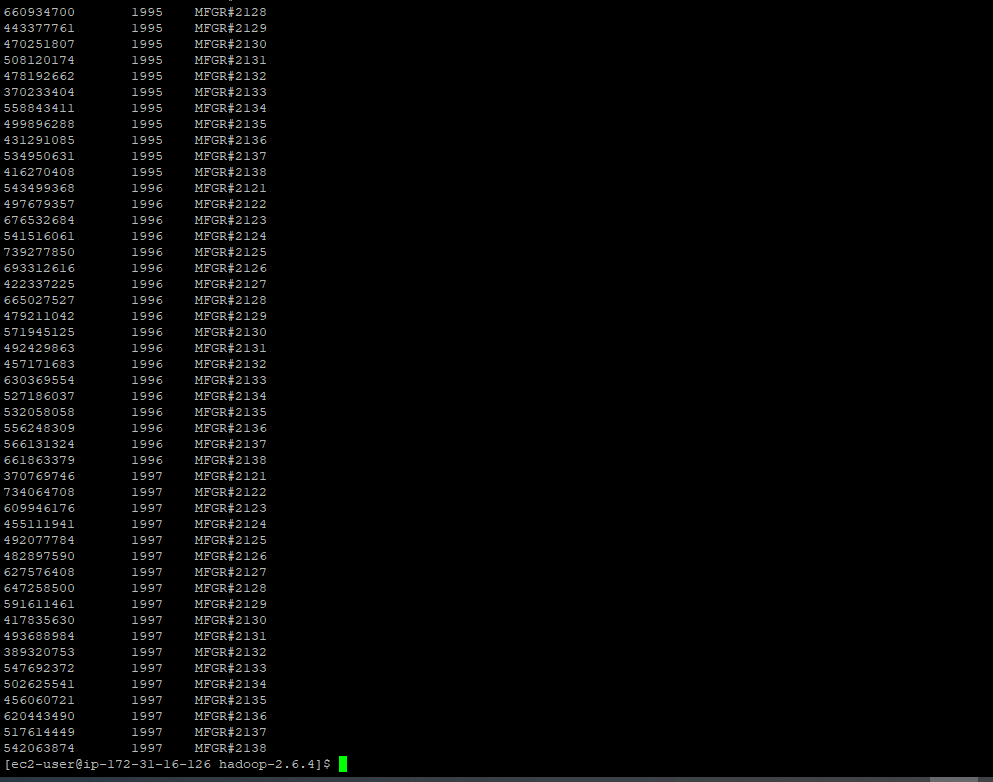


**Second Pass:**









**The code of the second pass:**

**result2Mapper.py**

****

**result2Reducer:**

****

# Part 3: Clustering

Using Hadoop streaming and randomly generated data (similar to what you did in Assignment6, but generate 2,100,000 rows and 6 columns of data) perform four KMeans iterations manually, using 4 centers**.** You can randomly choose the initial centers, such as by picking 4 random points from your data. For each of four KMeans iterations, include the centers produced by your code. Please do not submit the command line four times, without the corresponding output.

**Code used to generate the file:**

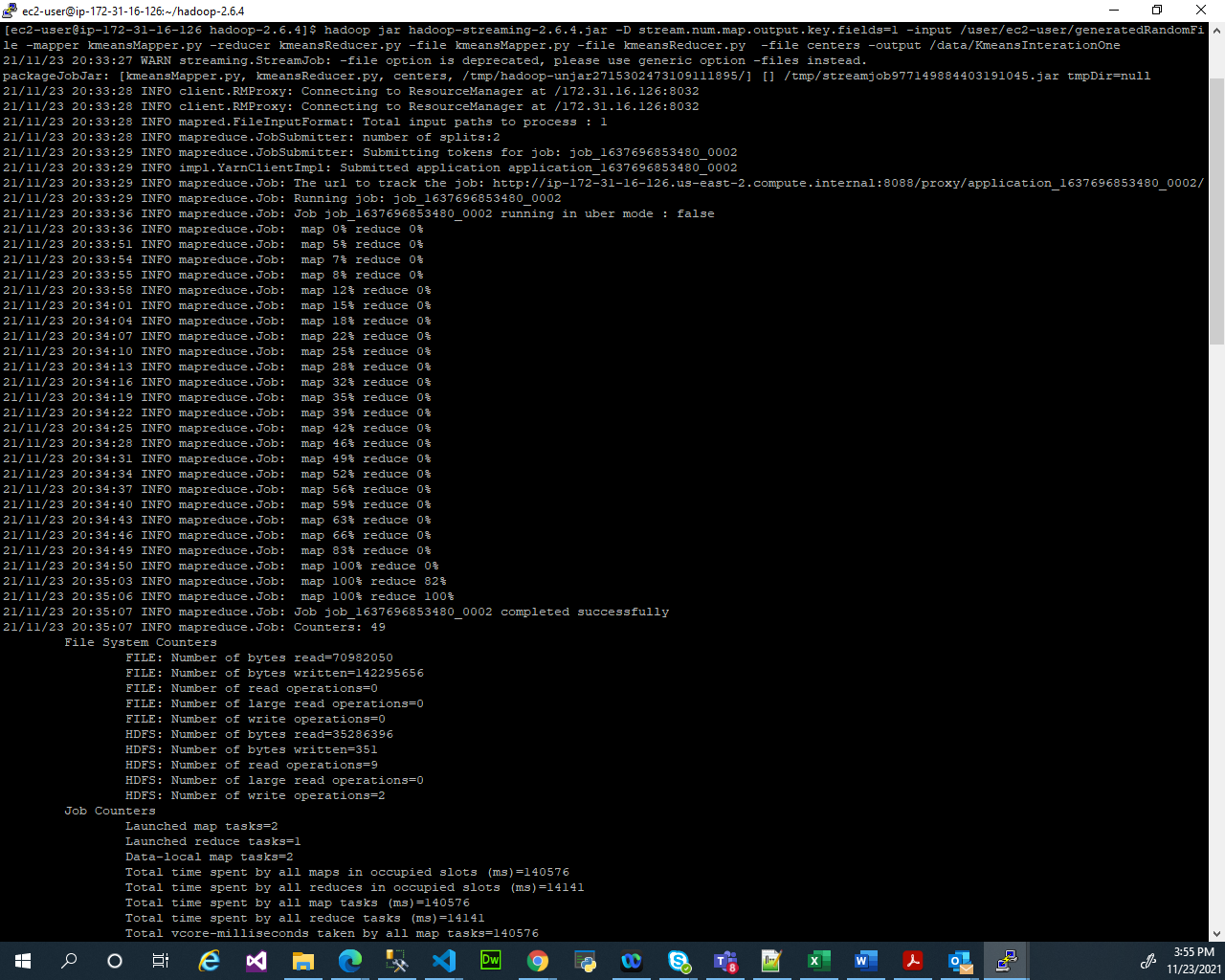
****

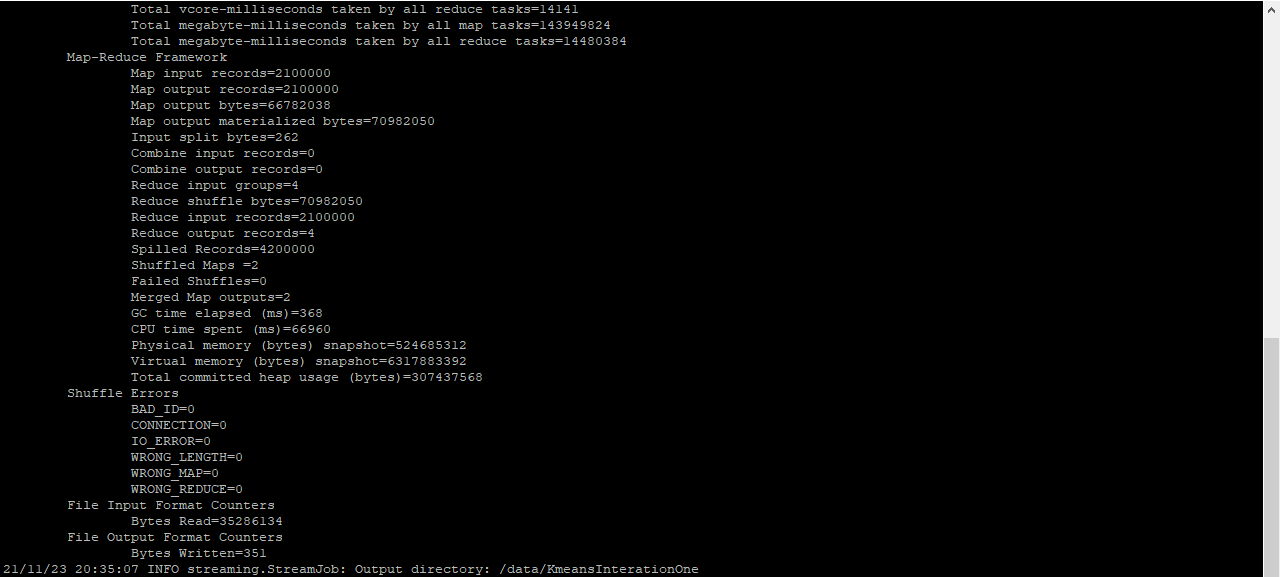
This would require passing a text file with cluster centers using -file option as discussed in class, opening the centers.txt in the mapper with open(‘centers.txt’, ‘r’) and assigning a key to each point based on which center is the closest to each particular point. Your reducer would then compute the new centers by averaging the points, which would conclude the iteration. At that point, the output of the reducer with new centers can be given to the next pass of the same map reduce code using the -file option (you would need to get the output from HDFS into a local file for that).

The only difference between first and subsequent iterations is that in first iteration you have to pick the initial centers. Starting from the 2nd iteration, the centers will be given to you by a previous pass of KMeans, and so on. Include the centers you computed at each iteration in your answer.

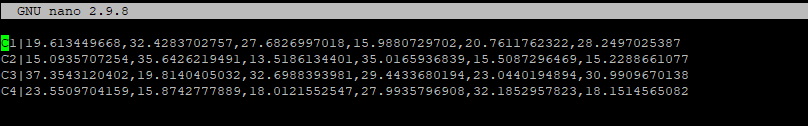
These were my initial centers I manually picked up:



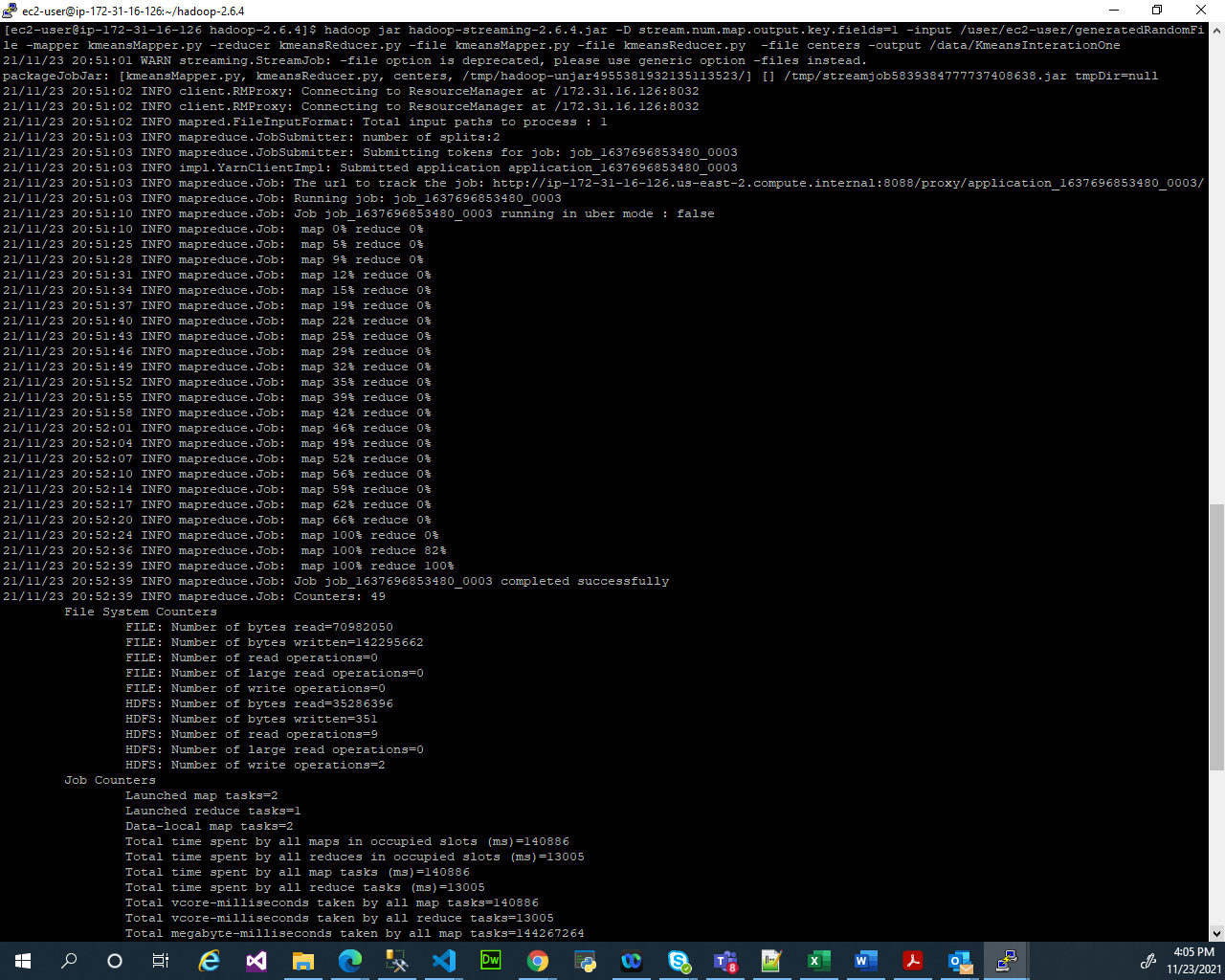


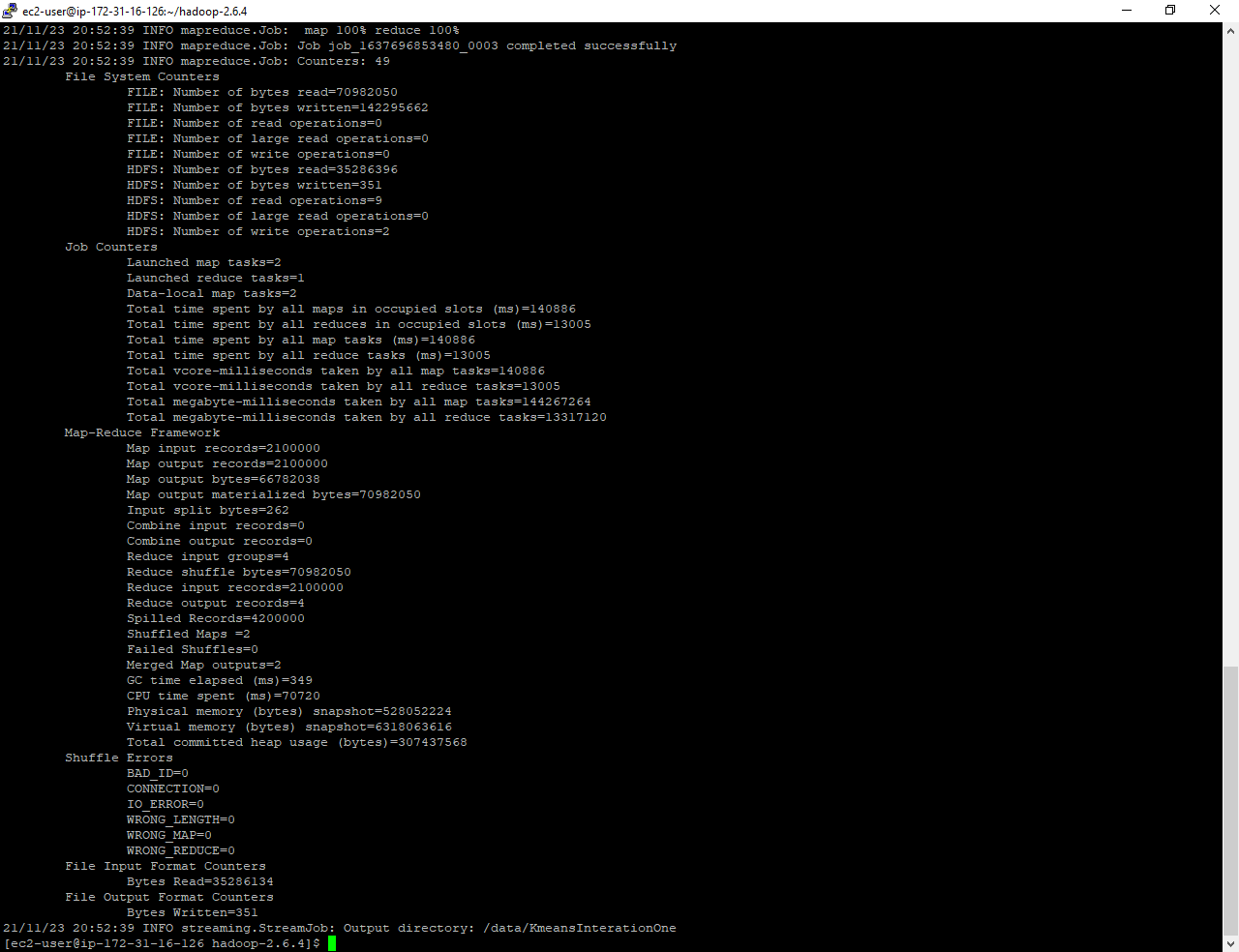


This produced the converged centers:

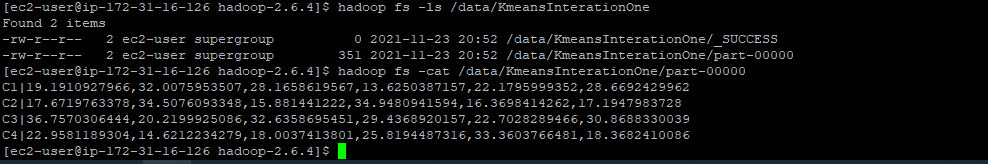


2nd run:

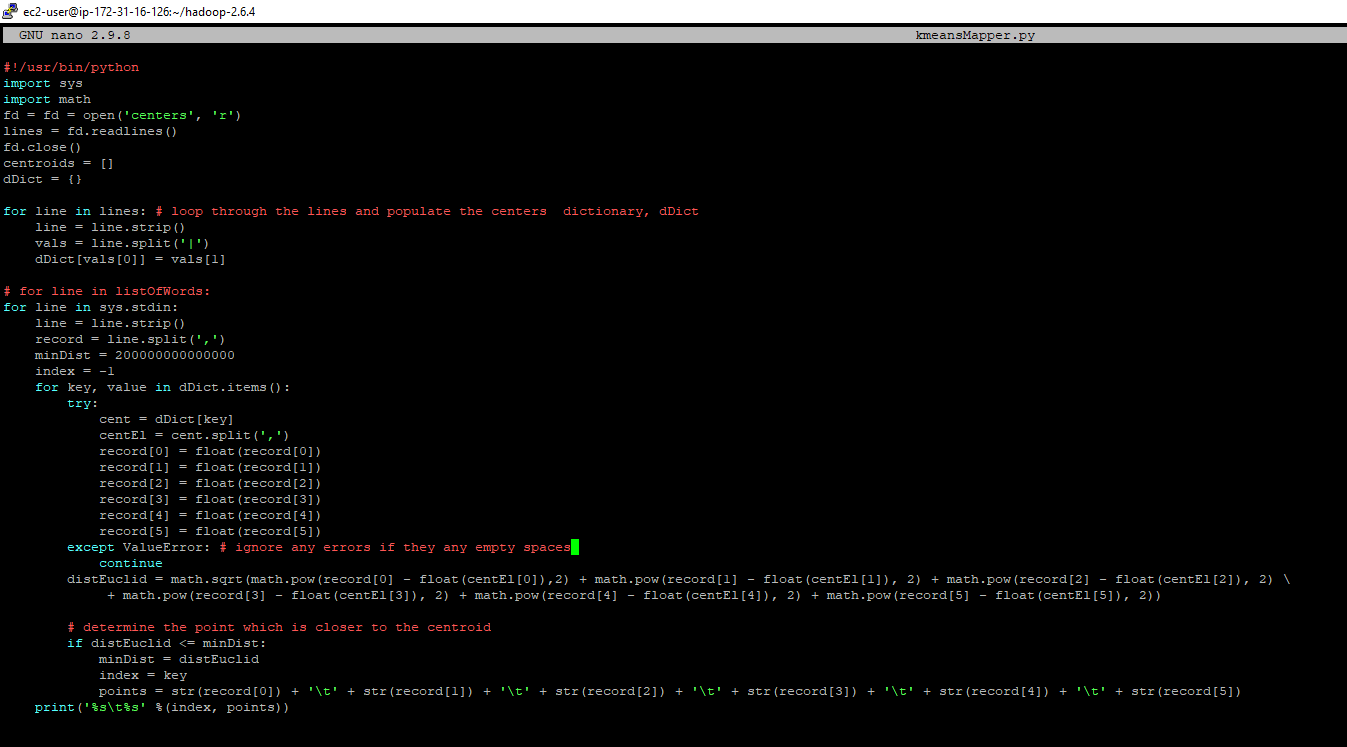




The centers in the /data/kmeansInterationOne:



The code used for the KmeansMapper.py and kmeansReducer.py

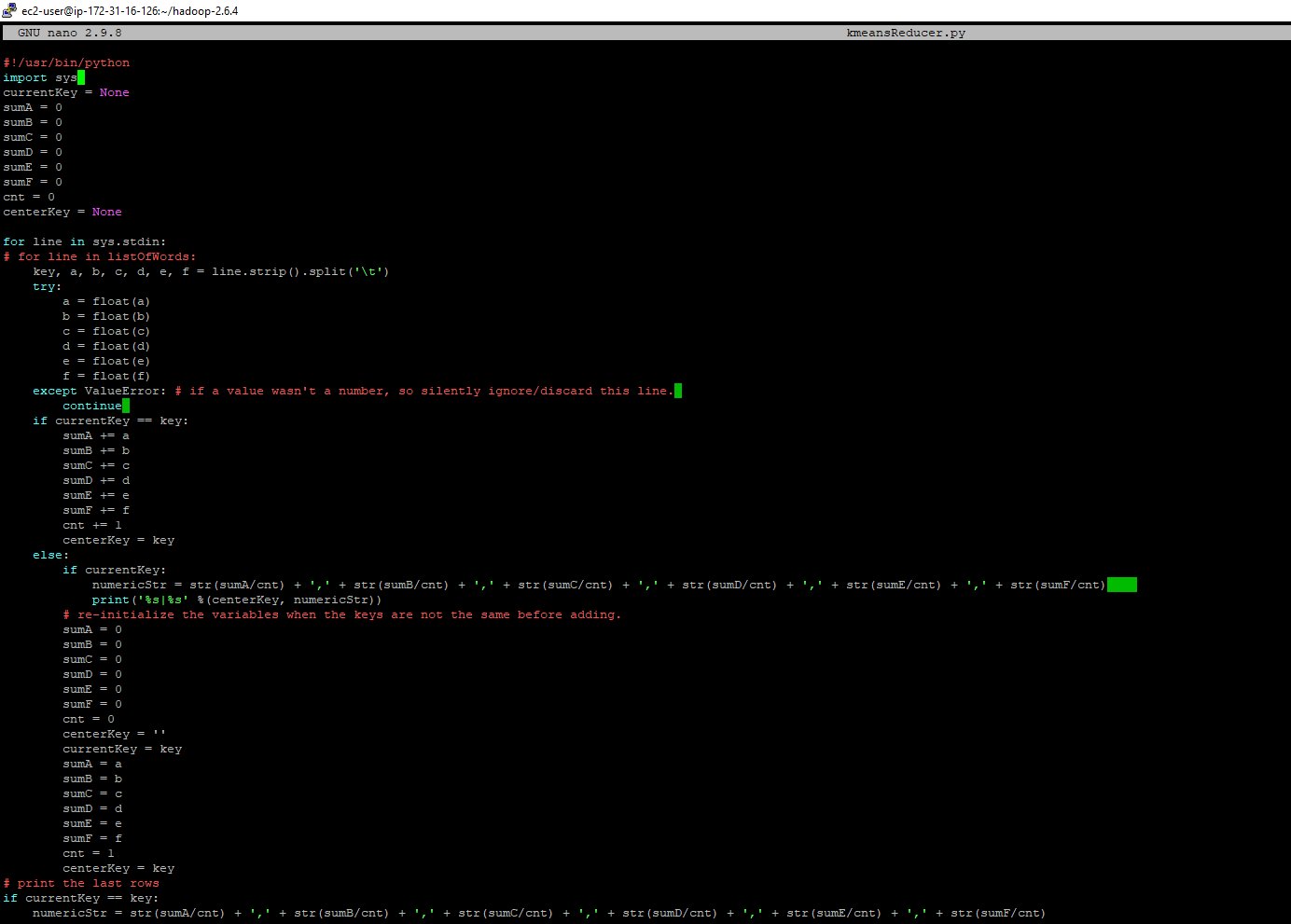


**KmeansMapper.py:**



kmeansReducer.py







Submit a single document containing your written answers. Be sure that this document contains your name and “CSC 555 Project Phase 2” at the top.